

V850ES/Fx3 and V850ES/Fx3-L Microcontrollers Usage Restrictions	Document No.	ZBG-BG-09-0048	1/4
	Date issued	November 9, 2009	
	Issued by	Naoto Iga, Group Manager Automotive Systems Division Microcomputer Operations Unit NEC Electronics Corporation	
Related documents: V850ES/Fx3 Hardware User's Manual: U17793EE2V2UM00 (2nd edition) V850ES/FE3-L, V850ES/FF3-L, V850ES/FG3-L Hardware User's Manual: U18743EE1V2UM00 (1st edition) V850ES Architecture User's Manual: U15943EJ3V0UM00 (3rd edition)	Notification classification	√	Usage restriction
			Upgrade
			Document modification
			Other notification

1. Affected products

All V850ES/Fx3 and V850ES/Fx3-L microcontrollers

For the names of the specific products in each series, see attachment 1.

2. Notification

The following restriction for the V850ES/Fx3 and V850ES/Fx3-L microcontrollers has been found.

No. 1 Restriction on executing a `mul` or `mulu` instruction

Description:

The following occur if a specific instruction sequence (sequence 1 or 2 below) is executed.

- The result of executing a multiplication instruction is not stored in the relevant general-purpose register.
- As a result of executing an `ld` instruction for a mis-aligned address, the data at an incorrect address is read and stored in the relevant general-purpose register.

Sequence 1:

In the following instruction sequence, the RAM is read by one of the instructions in (2) at the same time as the RAM is accessed by a DMA transfer:

- (1) `ld` or `sld`: A load instruction for the internal ROM
- (2) `ld` or `sld`: A load instruction for the internal RAM
- (3) `mul` or `mulu`: An instruction that multiplies word data and whose result is truncated to 32 bits^{Note 1}

... **Note 2**

- (4) `ld` or `sld`: A load instruction for a mis-aligned address in the internal ROM or RAM

Notes 1. For a `mul` or `mulu` instruction, the operation described in this restriction occurs if `r0` is specified for the third operand (`reg3`), or the same register is specified for the second operand (`reg2`) and third operand (`reg3`), as shown below:

```
mul reg1, reg2, reg3 (reg3 = r0 or reg2 = reg3)
```

```
mul imm9, reg2, reg3 (reg3 = r0 or reg2 = reg3)
```

```
mulu reg1, reg2, reg3 (reg3 = r0 or reg2 = reg3)
```

```
mulu imm9, reg2, reg3 (reg3 = r0 or reg2 = reg3)
```

For a program written in C, the CA850 does not generate the `mul` and `mulu` instructions if the same register is specified for the `reg2` and `reg3` operands.

2. This restriction applies if it takes 2 clock cycles or less between the instructions in (3) and (4).

This restriction does not apply in any of the following cases:

- DMA is not used to transfer data to or from the internal RAM.
- The data read by the load instruction in (1) is used to specify the target address of the load instruction in (2).
- The data read by the load instruction in (2) is referenced by the multiplication instruction in (3).
- The data obtained by the multiplication instruction in (3) is used to specify the target address of the load instruction in (4).
- The data obtained by the multiplication instruction in (3) is referenced by an instruction executed between the instructions in (3) and (4).
- At least one of the following instructions is executed between the instructions in (3) and (4):
 - A multiplication instruction (`mul`, `mulh`, `mulhi`, `mulu`)
 - A bit manipulation instruction (`clr1`, `not1`, `set1`, `tst1`)
 - A special instruction (`callt`, `dispose`, `switch`)
- The instruction in (4) is a load instruction that accesses the memory in bytes (`ld.b`, `ld.bu`, `sld.b`, or `sld.bu`).
- The instructions in (1) to (4) are located in an external memory or the internal RAM.

Sequence 2:

In the following instruction sequence, access by the instruction in (1) ends at the same time as the instruction in (2) accesses the internal RAM:

(1) `ld` or `sld`: A load instruction for an external memory, a CAN peripheral I/O register, or the data flash area

... **Note 1**

(2) `ld` or `sld`: A load instruction for the internal RAM

(3) `mul` or `mulu`: An instruction that multiplies word data and whose result is truncated to 32 bits^{Note 2}

... **Note 3**

(4) `ld` or `sld`: A load instruction for a mis-aligned address in the internal ROM or RAM

- Notes 1.** This restriction applies if a different instruction is executed between the load instructions in (1) and (2), and the access by the instruction in (1) ends at the same time as the instruction in (2) accesses the internal RAM.
- 2.** For a `mul` or `mulu` instruction, the operation described in this restriction occurs if `r0` is specified for the third operand (`reg3`), or the same register is specified for the second operand (`reg2`) and third operand (`reg3`), as shown below:
- ```
mul reg1, reg2, reg3 (reg3 = r0 or reg2 = reg3)
mul imm9, reg2, reg3 (reg3 = r0 or reg2 = reg3)
mulu reg1, reg2, reg3 (reg3 = r0 or reg2 = reg3)
mulu imm9, reg2, reg3 (reg3 = r0 or reg2 = reg3)
```
- For a program written in C, the CA850 does not generate the `mul` and `mulu` instructions if the same register is specified for the `reg2` and `reg3` operands.
- 3.** This restriction applies if it takes 2 clock cycles or less between the instructions in (3) and (4).

This restriction does not apply in any of the following cases:

- An external bus, and CAN controller, and data flash area are not used.
- The data read by the load instruction in (1) is used to specify the target address of the load instruction in (2).
- The data read by the load instruction in (1) is referenced by an instruction executed between the instructions in (1) and (2).
- The data read by the load instruction in (2) is referenced by the multiplication instruction in (3).
- The data obtained by the multiplication instruction in (3) is used to specify the target address of the load instruction in (4).
- The data obtained by the multiplication instruction in (3) is referenced by an instruction executed between the instructions in (3) and (4).
- At least one of the following instructions is executed between the instructions in (3) and (4):
  - A multiplication instruction (`mul`, `mulh`, `mulhi`, `mulu`)
  - A bit manipulation instruction (`clr1`, `not1`, `set1`, `tst1`)
  - A special instruction (`callt`, `dispose`, `switch`)
- The instruction in (4) is a load instruction that accesses the memory in bytes (`ld.b`, `ld.bu`, `sld.b`, or `sld.bu`).
- The instructions in (1) to (4) are located in an external memory or the internal RAM.

Workaround:

Action for systems being developed and to be developed in future:

We regard this as a restriction related to the CPU's features and do not intend to correct the microcontroller. Instead, the CA850 compiler will be modified to automatically prevent instructions to which this restriction applies from being generated. This workaround does not work for instructions in assembly code, so the CA850 outputs a message for such instructions. How NEC Electronics will provide the update for the compiler is shown below.

If you are using another compiler, contact an NEC Electronics sales representative.

- If your compiler is the NEC Electronics compiler CA850:

The CA850 will be upgraded to add the above workaround.

The following table shows the version and release schedule for the C compiler package CA850, and the software package SP850 that includes the CA850.

| Product Name | Current CA850 Version | Version of CA850 After Upgrade | Language | Release Schedule |
|--------------|-----------------------|--------------------------------|----------|------------------|
| SP850, CA850 | V3.40                 | V3.42                          | Japanese | November 9, 2009 |
|              |                       |                                | English  | November 9, 2009 |

Action for already-developed systems:

Use the interview sheet in attachment 3 to check whether the restriction applies.

Application of this restriction to embedded software products:

This restriction applies to NEC Electronics real-time OSs and middleware as follows:

- Real-time OSs:

RX850: The restriction does not apply.

RX850 Pro: The restriction does not apply.

RX850V4: The restriction does not apply.

- Middleware:

GOFAST: The restriction does not apply.

JPEG: The restriction does not apply.

For products other than the above, contact an NEC Electronics sales representative. For third-party products, contact the vendor of the product.

Modification:

The device will not be corrected, so regard this restriction as a specification.

Use the latest compiler to avoid this restriction.

### 3. Development environment required to work around this restriction

Use NEC Electronics compiler CA850 V3.42 or later.

Note that the outputs from the above version might differ from the outputs from the older versions.

If you are using another compiler, contact an NEC Electronics sales representative.

**V850ES/Fx3 and V850ES/Fx3-L Microcontrollers**

## ▼ V850ES/Fx3

*μ*PD70F3370A, *μ*PD70F3371, *μ*PD70F3372, *μ*PD70F3373, *μ*PD70F3374, *μ*PD70F3375,  
*μ*PD70F3376A, *μ*PD70F3377A, *μ*PD70F3378, *μ*PD70F3379, *μ*PD70F3380, *μ*PD70F3381,  
*μ*PD70F3382, *μ*PD70F3383, *μ*PD70F3384, *μ*PD70F3385

## ▼ V850ES/Fx3-L

*μ*PDPD70F3610, *μ*PD70F3611, *μ*PD70F3612, *μ*PD70F3613, *μ*PD70F3614, *μ*PD70F3615,  
*μ*PD70F3616, *μ*PD70F3617, *μ*PD70F3618, *μ*PD70F3619, *μ*PD70F3620, *μ*PD70F3621,  
*μ*PD70F3622

### V850ES/Fx3 and V850ES/Fx3-L Microcontroller Usage Restrictions

| No. | Restriction                                                                  | Nickname                   | Product      | Note | Development Environment in Which the Restriction Can Be Worked Around |
|-----|------------------------------------------------------------------------------|----------------------------|--------------|------|-----------------------------------------------------------------------|
| 1   | Restriction on executing a <code>mul</code> or <code>mulu</code> instruction | V850ES/Fx3<br>V850ES/Fx3-L | All products | △    | Category: Compiler<br>Product name: CA850<br>Version: V3.42 or later  |

**Note** Whether or not the restriction applies

**Remark** The meaning of each symbol for **Note** is as follows:

- : Restriction does not apply
- : Restriction is already corrected
- ×: Restriction applies (correction is planned)
- △: Restriction applies (correction is not planned)

#### • First judgment: Judgment based on product usage conditions

Check the conditions under which you are using the product to see whether the restriction applies to the product. If the restriction might apply, perform a second judgment. If the restriction is judged to be not applicable, subsequent checking is not necessary.

##### ■ Checking the usage conditions

Select **Yes** or **No** for whether each feature from (1) to (4) below is used. If the product does not incorporate a feature, select **Not relevant** for the feature. If there is no item for which **Yes** is selected, the restriction does not apply.

(1) Data is not transferred to or from the internal RAM using **DMA**.

Yes  No  Not relevant

(2) An on-chip **CAN controller** is used.

Yes  No  Not relevant

(3) An on-chip **data flash**\* is used.

Yes  No  Not relevant

(4) An **external bus interface** is used.

Yes  No  Not relevant

\* Data flash is a flash memory provided only for storing data and therefore cannot execute programs.

• **Second judgment: Judgment based on compiling conditions**

**Interview Sheet (for second judgment)**

Use the following flowchart to judge whether the restriction applies. If the restriction might apply, a third judgment using check tools is required.

